Appeal No. 2017-2145

United States Court of Appeals

for the

Federal Circuit

CISCO SYSTEMS, INC.,

Plaintiff-Appellant,

– v. –

ARISTA NETWORKS, INC.,

Defendant-Appellee.

ON APPEAL FROM THE UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF CALIFORNIA IN
CASE NO. 5:14-CV-05344-BLF, JUDGE BETH LABSON FREEMAN

BRIEF FOR COPYRIGHT LAW PROFESSOR PAMELA SAMUELSON AS *AMICUS CURIAE* IN SUPPORT OF DEFENDANT-APPELLEE AND AFFIRMANCE

ROBERT WALKER
CATHERINE CRUMP
SAMUELSON LAW, TECHNOLOGY
& PUBLIC POLICY CLINIC
University of California, Berkeley,
School of Law
334 Boalt Hall (North Addition)
Berkeley, California 94720
(510) 664-4875

Counsel of Record for *Amicus Curiae*

December 22, 2017

UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT

Cisco Systems, Inc. v. Arista Networks, Inc.

No. 2017-2145

CERTIFICATE OF INTEREST

Pursuant to Federal Rule of Appellate Procedure 26.1 and Federal Circuit Rule 47.4, counsel for *amicus curiae* Copyright Law Professor Pamela Samuelson certifies the following:

1. The full name of the *amicus curiae* represented by me is: PAMELA SAMUELSON.

2. The name of the real parties in interest represented by me is: N/A.

3. All parent corporations and any publicly held companies that own 10% or more of stock in the party or *amicus curiae* represented by me are: NONE.

4. The names of all law firms and the partners or associates that appeared for the party or *amicus curiae* now represented by me in the trial court or are expected to appear in this Court (and who have not or will not enter an appearance in this case) are: NONE.

5. The title and number of any case known to counsel to be pending in this or any other court or agency that will directly affect or be directly affected by this court's decision in the pending appeal: NONE.

Dated: December 22, 2017

Respectfully submitted,
/s/ Robert Walker

Robert Walker
SAMUELSON LAW, TECHNOLOGY& PUBLIC POLICY CLINIC
University of California, Berkeley, School of Law
334 Boalt Hall (North Addition)
Berkeley, California 94720
(510) 664-4875
Counsel of Record for Amicus Curiae Copyright Law Professor Pamela Samuelson

TABLE OF CONTENTS

Page

TABLE OF AUTHORITIES

Page(s)

CASES

STATUTES

OTHER AUTHORITIES

STATEMENT OF INTEREST[1]

Pamela Samuelson is the Richard M. Sherman Distinguished Professor of Law at the University of California, Berkeley, School of Law, where she teaches and writes about copyright law.[2] Since the early 1980s, she has authored more than 100 law review articles, many of which focus on software copyright issues. She is a coauthor with Professors Mark A. Lemley, Robert P. Merges, and Peter Menell of a casebook on Software and Internet Law. Samuelson has also written a comparable number of articles for computing and scientific journals such as *Communications of the ACM*, for which she has been a contributing editor since 1990. She is also a Fellow of the Association for Computing Machinery (ACM), a computing professionals society with more than 100,000 members worldwide, and a member of the ACM Council. As this case will significantly impact software copyright law and in consideration of this Court's more limited experience with cases involving copyright relative to the vast number of patent appeals it has

[1] This brief is filed pursuant to Fed. R. App. P. 29(a) with the consent of all parties. Pursuant to Fed. R. App. P. 29(a)(4), *amicus* hereby states that none of the parties to this case nor their counsel authored this brief in whole or in part; no party or any party's counsel contributed money intended to fund preparing or submitting the brief; and no one else other than *amicus* and its counsel contributed money that was intended to fund preparing or submitting this brief.

[2] Professor Samuelson represents no institution, group, or association and has no personal interest at stake in the outcome of this case; institutional affiliations are for identification purposes only.

adjudicated, Professor Samuelson respectfully submits this brief *amicus curiae* to offer her expertise to aid the Court.

SUMMARY OF ARGUMENT

The scenes a faire doctrine, which recognizes elements of a work that are stock, rudimentary, or that arise naturally from a particular theme or setting, is a flexible and capacious limitation on copyright protection that can shield defendants from copyright liability. There are at least five senses in which the scenes a faire doctrine may apply in software copyright cases, including when infringement claims are based on the defendant's reuse of program command terms. First, program command terms can be scenes a faire insofar as they incorporate official industry standard terms. Second, program commands that are common or stock elements likely to be found in software of that kind may be scenes a faire elements. Third, program commands can also be scenes a faire if they logically flow from the functions to be performed. Fourth, program command terms can be scenes a faire if external factors, including market expectations, constrain programmers' choices of command terms. Fifth, under controlling Ninth Circuit precedents, program commands, like other user interface elements, can, by virtue of longstanding use in an industry, become standards in that industry, and hence scenes a faire elements.

Insofar as the jury heard evidence that Cisco's command-line interface ("CLI") terms were standards in one or more of these senses, it could reasonably have concluded that so many elements of the claimed compilation of terms were scenes a faire that Arista's use of those terms did not infringe. Such a finding is especially likely and appropriate given that compilations of program command terms are generally more functional than expressive. In some cases, program command terms, like functional compilations more generally, have been adjudged too functional to be protectable by copyright law. Some functional compilations, though protectable, have enjoyed a very thin scope of protection. This Court should be skeptical of Cisco's claim that the CLI elements used by Arista are in themselves a protectable work of authorship that Arista infringed, because those commands were subsets of a larger set of command terms and many were scenes a faire elements.

ARGUMENT

I. Scenes a Faire Elements of Computer Program Interfaces Are Unprotectable by Copyright Law.

The scenes a faire doctrine has often been applied in computer software cases to limit the scope of copyright protection.[3] The leading case on this point is *Computer Associates International, Inc. v. Altai, Inc.*, 982 F.2d 693 (2d Cir. 1992). The *Altai* decision identified five categories of program elements that should be regarded as unprotectable by copyright law under the scenes a faire doctrine:

> (1) the mechanical specifications of the computer on which a particular program is intended to run; (2) compatibility requirements of other programs with which a program is designed to operate in conjunction; (3) computer manufacturers' design standards; (4) demands of the industry being serviced; and (5) widely accepted programming practices within the computer industry.

[3] The District Court instructed the jury to decide first whether Arista had infringed Cisco's copyright and then whether Arista had proven an affirmative defense that the CLIs at issue were scenes a faire. *See* Appx5-6. It is, however, the plaintiff's burden in a copyright case to prove that the defendant took protectable expression from its work. *See* Appx5; 3 PAUL GOLDSTEIN, GOLDSTEIN ON COPYRIGHT § 10.5.1.2 at 10:94.1 (3d ed. 2005 & Supp. 2017) ("The abstraction-filtration-comparison test places a special burden on the copyright [owner] to highlight the original and expressive elements that it claims are infringed."). Scenes a faire elements are, by definition, not expressions that copyright can protect. Hence, it was error for the District Court to charge the jury to make a finding of infringement first and then to assess Arista's scenes a faire defense. If what the defendant took were scenes a faire elements, there was no infringement. *See, e.g.*, *Cavalier v. Random House, Inc.*, 297 F.3d 815, 824 (9th Cir. 2002) (concluding that similarities in books as to scenes a faire elements could not support a finding of copyright infringement).

Id. at 709-10 (citations omitted). The Second Circuit directed courts to filter out these unprotectable elements before engaging in the comparison stage of infringement analysis in computer program cases. *Id*. at 707-10.

Like the Second Circuit in *Altai*, the Federal Circuit has recognized the applicability of the scenes a faire doctrine in software copyright cases, most recently in *Oracle America, Inc. v. Google Inc.*, 750 F.3d 1339, 1363-64 (Fed. Cir. 2014). That doctrine "operates to bar certain otherwise creative expression from copyright protection," especially when such expressions are "standard, stock, or common to a topic, or if they necessarily follow from a common theme or setting," or if they are "naturally associated with the treatment of a given idea." *Id*. at 1363 (citations omitted). In addition, the Federal Circuit has acknowledged that "the scene a faire doctrine denies protection to program elements that are dictated by external factors…or 'widely accepted programming practices within the computer industry.'" *Id*. (citations omitted).

A. Command Terms That Embody Official Standards Are Unprotectable Under the Scenes a Faire Doctrine.

The District Court identified numerous clusters of official standard terms that Cisco embodied in the CLI when denying that firm's motion for a judgment as a matter of law. Appx9-11. This evidence supports the jury's verdict. It does not matter that the CLI elements on which Cisco bases its infringement claim did not, as a whole, consist of officially adopted standard terms. Because there is more than

one sense in which CLI commands can be scenes a faire elements, the jury could have considered the officially adopted standards as one type of scenes a faire elements, while other CLI terms were scenes a faire in other senses.

The existence of officially adopted standards was a significant factor affecting copyright scope in *Kohus v. Mariol*, 328 F.3d 848 (6th Cir. 2003). In considering a claim of infringement of a drawing of a latch for a children's portable play yard, the Sixth Circuit in *Kohus* recognized that "standard industry practices for constructing latches, or safety standards established by organizations like the American Society for Testing Materials and the Juvenile Products Manufacturer's Association" were "external considerations" that might limit the scope of the plaintiff's copyright. *Id.* at 856. Official standards were also significant considerations in *Secure Services Technology, Inc. v. Time & Space Processing, Inc.*, 722 F. Supp. 1354 (E.D. Va. 1989). The court granted the defendant's motion for summary judgment because a T-30 protocol imposed significant constraints on the design of interfaces for secure fax machines for military communications. SST argued that its variations in the implementation of this protocol were original enough to be copyrightable. However, the court found SST's variations to be too minor to qualify for copyright protection. *Id*. at 1362-63. Moreover, use of those variations was necessary for TSP's machines to work with SST machines, as military customers of these machines demanded. *Id*. at 1358.

The right of second comers to reproduce officially adopted standards was also recognized in cases such as *Veeck v. Southern Building Code Congress International, Inc.*, 293 F.3d 791 (5th Cir. 2002) (denying copyright protection to a privately drafted standard adopted as law) and *Building Officials & Code Administrators Int'l v. Code Technology, Inc.*, 628 F.2d 730 (1st Cir. 1980) (vacating preliminary injunction because of doubts about the copyrightability of a model code adopted by the state).

B. Common Names for Command Terms for Common Functions Are Unprotectable Under the Scenes a Faire Doctrine.

The unprotectability on scenes a faire grounds of common or expected elements of works of a particular kind was highlighted in a leading Ninth Circuit precedent, *Satava v. Lowry*, 323 F.3d 805 (9th Cir. 2003). Satava sued Lowry for infringement because of many visual similarities between its glass jellyfish sculpture and those made by Lowry. The Ninth Circuit reversed the trial court's issuance of a preliminary injunction because "no copyright protection may be afforded to the idea of producing a glass-in-glass jellyfish sculpture or to elements of expression that naturally follow from the idea of such a sculpture." *Id*. at 810.

The common-element type of scenes a faire has been recognized in computer program copyright cases as well. This Court in *Hutchins v. Zoll Medical Corp.*, 492 F.3d 1377, 1385 (Fed. Cir. 2007), for example, held that common phrases for a computerized system for performing cardiopulmonary resuscitation

(CPR) were unprotectable by copyright law under the scenes a faire doctrine. This Court quoted approvingly from a District Court decision that Hutchins' use of 27 CPR phrases, such as "call for help" and "check breathing," were "entirely functional" and "standard." *Id*. at 1384-85. The Federal Circuit added that "[c]opyright does not protect individual words and 'fragmentary phrases' when removed from their form of presentation and compilation." *Id*. at 1385. *See also Brown Bag Software v. Symantec Corp.*, 960 F.2d 1465 (9th Cir. 1992) (similarities in user interfaces of two outlining programs due in part to use of standard components for works of that kind).

The common elements type of scenes a faire applies to program commands. It is necessary for users to invoke the functionality of interactive computer programs through use of command terms that typically identify common functions that the programs of that type were designed to perform. Command terms that are common names of common functions are scenes a faire and should also not serve as a basis of an infringement claim.

Insofar as CLI terms at issue were names or phrases representing common functions that developers of software for Ethernet switching technologies would need to perform, they are scenes a faire elements. "Boot system," for instance, is a term common in software systems for doing just what the name implies: making the system "boot" (that is, start up). *See* Appx3. "Show inventory" is likewise a

concise description of the name of that function. *Id*. To the extent that the jury heard evidence establishing linkages between command terms and device features, Appx9, this evidence would support a finding that those parts of the CLIs that were common for programs of that kind were unprotectable under the scenes a faire doctrine. Common or expected commands, coupled with official industry standard commands, would support Arista's scenes a faire defense.

C. Command Terms That Flow From Program Logic Are Unprotectable Under the Scenes a Faire Doctrine.

Some creative elements of computer programs naturally flow from the logic that the programmer devised in the course of developing its software. The Nimmer treatise, among other sources, recognizes that program logic is not protectable by copyright law. *See* 3 MELVILLE B. NIMMER & DAVID NIMMER, NIMMER ON COPYRIGHT § 13.03[F][2] (2017). *See also* U.S. COPYRIGHT OFFICE, COMPENDIUM OF U.S. COPYRIGHT OFFICE PRACTICES § 721.7 (3d ed. 2017) ("[T]he Office will not register the functional aspects of a computer program, such as the program's algorithm, formatting, functions, logic, system design, or the like."). When elements of a computer program such as command terms are the outgrowth of program logic, they too should be regarded as scenes a faire elements.

A good example of the logical-flow type of scenes a faire limit on copyright is *Auto Inspection Services, Inc. v. Flint Auto Auction, Inc.*, No. 06-15100, 2006 WL 3500868 (E.D. Mich. Dec. 4, 2006). That court denied the plaintiff's motion

for a preliminary injunction on its copyright claim because customer requirements "dictated exactly what the programs were to do and look like" and "other external factors such as efficiency, hardware standards, compatibility requirements, and ease of use seriously limit the ways in which [the] software can be written." *Id.* at *7. The defendant had consulted with vehicle inspectors "in order to determine the most logical, efficient, and user friendly way to write the software." *Id*. at *3. The on-screen displays needed to "conform to all industry requirements and follow the normal and logical flow of a vehicle inspection." *Id*.

The logical nature of user interface command terms was influential in thwarting the plaintiff's claim of copyright in a compilation of the terms in *MiTek Holdings, Inc. v. Arce Engineering Co.*, 89 F.3d 1548 (11th Cir. 1996). The Eleventh Circuit affirmed a trial court ruling for Arce primarily because the command structure at issue implemented the logical step-by-step process that wood truss designers would ordinarily follow. *Id*. at 1556-58. The court also noted that the look of the MiTek program was an "industry standard" for computer-aided design programs. *Id*. at 1557, n.20. *See also Hutchins*, 492 F.3d at 1384 (affirming a trial court's denial of copyright claim in a "system of logic whereby CPR instructions are provided by computerized display"); *Apple, Inc. v. Microsoft Corp.*, 35 F.3d 1435, 1439 (9th Cir. 1994) (scenes a faire if user interface elements "arise from the use of common ideas or their logical extension").

Any elements of the Cisco's CLI that naturally flowed from the logic of the functions that the Cisco technology was designed to perform should also be considered scenes a faire elements.

D. External Factors, Including Consumer Expectations, Can Render Some Command Terms Unprotectable Under the Scenes a Faire Doctrine.

Consumer expectations may also influence what will be considered scenes a faire elements of protected works, as the Second Circuit recognized in *Zalewski v. Cicero Builder Developer, Inc.*, 754 F.3d 95 (2d Cir. 2014). Style elements associated with particular types of architectural designs (e.g., neoclassical homes) are common-element types of scenes a faire, but in addition, the doctrine applies to "certain market expectations for homes or commercial buildings. Design features used by all architects, because of consumer demand, also get no protection." *Id*. at 105.

The external factors type of scenes a faire was important to the Tenth Circuit's ruling in *Mitel, Inc. v. Iqtel, Inc.*, 124 F.3d 1366 (10th Cir. 1997). In rejecting Mitel's copyright claim based on Iqtel's reuse of command codes for the latter's competing telecommunications product, the court observed that "much of the expression in Mitel's command codes was dictated by the proclivities of technicians and limited by significant hardware, compatibility, and industry requirements." *Id*. at 1375. In addition, hardware compatibility and telephone

network capabilities contributed to the court's conclusion that Mitel command codes for accessing features of a telecommunications system were scenes a faire elements. *Id.* at 1375-76. *See also Plains Cotton Coop. Ass'n v. Goodpasture Computer Serv., Inc.*, 807 F.2d 1256, 1262 (5th Cir. 1987) (recognizing that market factors, including prospective user community expectations, may play a significant role in determining the sequence and organization of a program's user interface).

Consumer expectations were also significant considerations in persuading the First Circuit that Borland should not be held liable for copyright infringement for its reuse of the same command structure as Lotus 1-2-3 in *Lotus Development Corp. v. Borland International, Inc.*, 49 F.3d 807 (1st Cir. 1994), *aff'd by an equally divided Court*, 516 U.S. 233 (1996). To attract users to switch from the Lotus spreadsheet program to Borland's award-winning software, Borland developed an emulation interface that featured the same command structure as 1-2-3. It did so to accommodate two interests of prospective customers: first, it enabled users who had constructed macros in 1-2-3 for commonly performed sequences of functions to migrate those macros to Borland's program, and second, it enabled users who had invested in learning the Lotus commands to continue to draw upon their knowledge when using Borland's product.

The First Circuit characterized as "absurd" Lotus's theory that users should have to learn a different command hierarchy to perform the same operations in a

different way for each program they used. *Id*. at 817-18. User investments in macros also favored Borland's defense:

> Under the district court's holding, if the user wrote a macro to shorten the time needed to perform a certain operation in Lotus 1-2-3, the user would be unable to use that macro to shorten the time needed to perform that same operation in another program. Rather, the user would have to rewrite his or her macro using that other program's menu command hierarchy. This is despite the fact that the macro is clearly the user's own work product.

Id. at 818. Judge Boudin concurred, finding it "hard to see why customers who have learned the Lotus menu and devised macros for it should remain captives of Lotus because of an investment in learning made by the users and not by Lotus." *Id*. at 821 (Boudin, J., concurring).

To the extent that some of Cisco's CLI terms were constrained by similar external factors, they too can be considered scenes a faire.

E. User Interface Elements That Become Standards Over Time Are Unprotectable Under the Scenes a Faire Doctrine.

Although the District Court instructed the jury to find in favor of Arista's scenes a faire defense only if the CLI elements it used were scenes a faire at the time Cisco developed them, Appx6, Ninth Circuit case law recognizes that computer program user interface designs can become unprotectable standards over

time.[4] To the extent that the jury may have considered evidence that some elements of Cisco's CLI had become standard over time, this evidence would further support a finding of non-infringement on scenes a faire grounds.

The leading case on point is *Apple Computer, Inc. v. Microsoft Corp.*, 799 F. Supp. 1006 (N.D. Cal. 1992), *aff'd*, 35 F.3d 1435 (9th Cir. 1994). Microsoft proffered evidence that numerous elements of Apple's graphical user interface ("GUI") that Microsoft had incorporated into its Windows GUI had become standard in other firms' GUIs, most of which had been developed after Apple's GUI. *Apple*, 799 F. Supp. at 1023-24. For instance, the overlapping windows feature that Apple claimed as part of its original GUI expression was held unprotectable because twenty-three other programs created after Apple's program utilized that design. *Id*. at 1033. Also held unprotectable was the downsizing of icons because it was a "standard treatment in the industry." *Id*. at 1044. Indeed, the court included a "scenes a faire" table in its opinion showing how common were the key elements of the GUI that Apple wanted to claim as its own. *Id*. at 1023-24.

Similarities in functional elements of software GUIs, the court observed, "does not suggest unlawful copying, but standardization across competing products for functional considerations." *Id*. at 1023. Standardization of "visual features in a

[4] On this issue, this Court's *Oracle* decision, 750 F.3d at 1364, did not conform to the Ninth Circuit's conception of scenes a faire, which, as the *Apple v. Microsoft* case shows, *infra*, can happen over time.

computer's interface helps to achieve its purpose." *Id*. The court went on to say that "[s]ome visual displays are or become so closely tied to the functional purpose of the article that they become standard." *Id*.

The Ninth Circuit affirmed the District Court's ruling in *Apple* because that court had appropriately applied "the limiting doctrines of originality, functionality, standardization, scenes a faire, and merger to find that Microsoft had not copied protectable elements of the Apple GUI. *Apple*, 35 F.3d at 1438. The court regarded the District Court's decision as consistent with a "long line of copyright decisions which recognizes that works cannot be substantially similar where analytic dissection demonstrates that similarities in expression are either authorized or arise from the use of common ideas or their logical extensions." *Id*. at 1439.

The Goldstein treatise identifies *Sega Enterprises, Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), along with the Ninth Circuit's *Apple* decision, as supporting the proposition that program interfaces can become standards over time. 1 GOLDSTEIN, *supra*, § 2.3.2.1 at 2:40 n.46.[5] This position is also consistent with the view of the National Commission on New Technological Uses of Copyrighted

[5] Professor Goldstein regards the *Altai* decision as "hospitably inclined to the proposition that merger should be tested at the time the expression was copied rather than at the time it was created." 1 GOLDSTEIN, *supra*, § 2.3.2.1 at 2:40 n.47. The logic of Goldstein's assessment on merger would apply equally well to scenes a faire. This Court has recognized the kinship of the merger and the scenes a faire doctrines. *Hutchins*, 492 F.3d at 1385.

Works (CONTU) which recommended copyright protection for software. CONTU stated that "when specific instructions, *even though previously copyrighted*, are the only and essential means of accomplishing a given task, their later use by another will not amount to an infringement"). NAT'L COMM'N ON NEW TECHN. USES OF COPYRIGHTED WORKS, FINAL REPORT 20 (1979) (emphasis added).[6]

Thus, Ninth Circuit precedents, a prominent copyright scholar, and CONTU all support a scenes a faire verdict as to elements that became standards over time.

F. A Scenes a Faire Defense Could Be Based On a Combination of the Varieties of Scenes a Faire Elements.

This Section has demonstrated that courts have recognized many different types of scenes a faire elements. Most of the five types of scenes a faire elements discussed above would have been scenes a faire at the time Cisco adopted them for its CLI, but even those that became standards over time can rightly be regarded as scenes a faire. Under Ninth Circuit law, any and all scenes a faire elements should be filtered out before reaching a verdict of infringement. *Apple*, 35 F.3d at 1443-44. It is not possible to know how many of the CLI terms at issue that the jury regarded as scenes a faire elements. But insofar as there was substantial

[6] For a discussion of the merger doctrine in software cases, *see, e.g.*, Pamela Samuelson, *Functionality and Expression in Computer Programs: Refining the Tests for Software Copyright Infringement*, 31 BERKELEY TECH. L.J. 1215, 1270-84 (2016).

evidence of several different kinds of scenes a faire elements in the Cisco CLI terms at issue, this Court should uphold the jury's verdict of non-infringement.

Affirmance of the jury verdict is particularly appropriate because this Court has previously recognized that appellate courts should presume that juries made findings of fact that would support their verdicts. *See Kinetic Concepts, Inc. v. Smith & Nephew, Inc.*, 688 F.3d 1342, 1356-57 (Fed. Cir. 2012) ("We first presume that the jury resolved the underlying factual disputes in favor of the verdict [ ] and leave those presumed findings undisturbed if they are supported by substantial evidence") (citation omitted). *See also Hana Financial, Inc. v. Hana Bank*, 135 S. Ct. 907 (2015) (rejecting petitioner's argument that a jury verdict should be overturned as a matter of law in trademark case when the key issue was one of fact).

II. Copyright Protection Has Often Been Withheld From User Interface Commands and Other Types of Functional Compilations.

Numerous appellate court decisions have held that copyright protection does not extend to sets of user interface command terms. Because copyright law does not protect functions and commands are names of functions, courts have often been reluctant to require second comers to engage in needless variation by adopting different names for the various functions their programs perform. Competition and ongoing innovation, as well as consumer welfare, are fostered by allowing reuses of commands as long as second comers reimplement those functions in

independently written code. These rulings are in keeping with other decisions involving claims of copyright in functional compilations.

A. Sets of Program Commands Have Often Been Held Unprotectable by Copyright Law.

The leading Ninth Circuit case on the unprotectability of program command terms is *Ashton-Tate Corp. v. Ross*, 728 F. Supp. 597 (N.D. Cal. 1989), *aff'd*, 916 F.2d 516 (9th Cir. 1990). The District Court ruled that many command terms on Ross' list of commands for a spreadsheet program, which he claimed Ashton-Tate had used for the user interface of its software, were commonly found in programs of that sort. Ross had merely told Ashton-Tate "what tasks he believed the interface should allow the user to perform." *Ashton-Tate*, 728 F. Supp. at 602. The court consequently granted Ashton-Tate's motion for summary judgment. The Ninth Circuit affirmed the lower court's ruling that Ross' list of commands was an unprotectable idea under 17 U.S.C. § 102(b). *Ashton-Tate*, 916 F.2d at 521-22. As a result, it rejected Ross' claim that he was a joint author of that software. *Id*. at 520.

Two other appellate courts that ruled against copyright claims for reuses of program command structures were the First Circuit in *Borland* and the Eleventh Circuit in *MiTek*. The doctrinal basis for the First Circuit's ruling in *Borland* and the Eleventh Circuit's ruling in *MiTek*, like the Ninth Circuit in *Ashton-Tate*, focused on § 102(b) exclusions. *See* 17 U.S.C. § 102(b) ("In no case does

copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work."). The First Circuit regarded the Lotus 1-2-3 command structure as an unprotectable method of operating a spreadsheet program, *Borland*, 49 F.3d at 815. The Eleventh Circuit considered the command structure in *MiTek* as an embodiment of an unprotectable process. *MiTek*, 89 F.3d at 1556-57. These decisions implicitly recognize that computer programs are virtual machines and commands are merely the names of the functions these machines are capable of performing.

Although the Tenth Circuit in *Mitel* was unpersuaded by Iqtel's § 102(b) defense, it nevertheless decided against Mitel's copyright claim for Iqtel's reuse of program commands on lack of originality and scenes a faire grounds. *Mitel*, 124 F.3d at 1372-76. *See also Cisco Sys., Inc. v. Huawei Techs. Co.*, 266 F. Supp. 2d 551, 554 (E.D. Tex. 2003) (denying Cisco's motion for a preliminary injunction against Huawei's use of the Cisco CLI in its competing product because Cisco had failed to offer a filtration analysis to exclude unprotectable elements of its work, such as ideas, processes, facts, public domain information, and scenes a faire material).

Program commands and command structures have been held unprotectable on other grounds as well. Before a jury, Google prevailed with its fair use defense for its reuse of parts of the command structure of Oracle's Java API. Although Oracle moved for a judgment as a matter of law, the District Court denied this motion, identifying numerous issues of fact on which the jury heard conflicting testimony and concluding that the jury must have believed Google's evidence over Oracle's. *See Oracle Am., Inc. v. Google Inc.*, No. C 10-03561 WHA, 2016 WL 3181206 at *1 (N.D. Cal. June 8, 2016) (order denying Oracle's motion for judgment as a matter of law).[7]

B. Functional Compilations Have Often Been Held Unprotectable by Copyright.

These rulings in software command structure cases are consistent with numerous precedents involving copyright claims in other types of functional writings, such as lists of hardware parts and lists of configurations of uncopyrightable products such as keys and ball bearings. *See, e.g.*, *RBC Nice*

[7] Arista similarly prevailed at trial on its scenes a faire defense. In denying Cisco's motion for judgment as a matter of law, the District Court found substantial evidence in the record to support the jury's verdict. Appx7-17. Insofar as the record contains evidence that supports Arista's defense, appellate court deference to the jury's findings is appropriate. *See, e.g., Harper v. City of Los Angeles*, 533 F.3d 1010, 1021 (9th Cir. 2008) ("A jury's verdict must be upheld if it is supported by substantial evidence, which is evidence adequate to support the jury's conclusion, even if it is also possible to draw a contrary conclusion.") (citation omitted).

Bearings, Inc. v. Peer Bearing Co., 676 F. Supp. 2d 9 (D. Conn. 2009) (denying claim of copyright in compilation of load data for ball bearings); *Windgate Software, L.L.C. v. Minnesota Computers, Inc.*, 504 F. Supp. 2d 582 (D. Minn. 2007) (denying preliminary injunction based on defendant's copying of more than six thousand IBM part numbers and part descriptions from plaintiff's database); *Continental Micro, Inc. v. HPC, Inc.*, No. 95 C 3829, 1997 WL 309028 (N.D. Ill. June 4, 1997) (ruling against copyright claim in compilation of data about the configuration of keys); *Sinai v. Bureau of Automotive Repair*, No. C-92-0274-VRW, 1992 WL 470699 (N.D. Cal. Dec. 21, 1992) (finding against claim of copyright in compilation of information about an auto emission control system).[8]

Even when functional compilations are deemed copyrightable, the scope of protection they enjoy is "thin" because functional contents must be filtered out before assessing infringement claims. *See, e.g.*, *Apple*, 35 F.3d at 1442; *Decorative Aides Corp. v. Staple Sewing Aides Corp.*, 497 F. Supp. 154, 157-58 (S.D.N.Y. 1980) (instructions for making drapes with sewing aide were copyrightable, but not infringed by defendant's very similar instructions). This means that no infringement can be found unless the two works at issue are virtually identical.

[8] These and other functional writing cases are discussed in Pamela Samuelson, *Functional Compilations*, 54 HOUSTON L. REV. 321, 335-54 (2016) (identifying four types of functionality that courts have identified in denying compilation copyright claims).

Apple, 35 F.3d at 1439-40. Insofar as the jury could have found that Cisco's and Arista's software were not virtually identical, a jury verdict for Arista should not be disturbed.

III. Cisco and Its Amici Mischaracterize the Work(s) of Authorship at Issue in This Case.

Cisco and its amici have sought to define the work of authorship at issue in this case as certain CLI elements that Arista used in its competing software. *Cisco Sys. Inc. v. Arista Networks, Inc.*, No. 14-cv-05344-BLF, 2016 WL 4440239, at *2 (N.D. Cal. Aug. 23, 2016); Br. for Amici Curiae The Mathworks, Inc., et al. at 5-6. This is an erroneous conception of the relevant work(s). The District Court noted that Cisco's copyright complaint comprised twenty-six registered versions of Cisco's four Internetwork operating system (IOS) programs embedded in Cisco's technologies, as well as associated user manuals. Appx1328. Because the jury found that user manual copyrights were not infringed, Appx1430, the relevant works are the IOS programs.

The District Court properly rejected Cisco's claim that the CLIs at issue were the relevant work, *Cisco*, 2016 WL 4440239 at *2-3, but decided that the work(s) at issue was Cisco's user interface. Appx5. This characterization of the work(s) is incorrect. User interfaces of computer programs are not works that are separately copyrightable from the programs in which they are embodied. They are non-literal elements of those programs that may sometimes be expressive enough

to be copyright-protectable, as the Ninth Circuit acknowledged in *Apple*, 35 F.3d at 1445-46. But even when copyright-protectable, these program interfaces typically enjoy a very thin scope of such protection because they are more the product of functional than expressive creativity. *Id*. at 1442. *See also ATCS Int'l LLC v. Jefferson Contracting Corp.*, 807 F. Supp. 2d 516 (E.D. Va. 2011) (compilation of technical drawings likely unprotectable by copyright law because they were the product of skilled engineering, not expressive creativity).

Although the Copyright Act of 1976 does not define the meaning of the term "work," courts usually rely on what the copyright registration certificate says the protected work is. Paul Goldstein, *What Is a Copyrighted Work?*, 58 UCLA L. REV. 1175, 1175, 1177 (2010). Professor Goldstein has pointed out that copyright owners sometimes have incentives to "skew the calibration of [the] work toward a size more granular than principle, law, or practice might reasonably condone." *Id*. at 1176. "Giving authors too great a say in establishing their work's boundaries may, it might be feared, incite self-seeking claimants to turn these boundaries to an advantage that unfairly taxes copiers and, therefore, the public." *Id*. at 1176-77. *See also* Margot E. Kaminski & Guy A. Rub, *Copyright's Framing Problem*, 64 UCLA L. REV. 1102, 1105-10 (2017) (pointing out that courts sometimes "zoom in" to characterize as the protected work a portion of the work and sometimes "zoom out" to treat more than one work as the work). By trying to characterize the

work(s) at issue as the CLI commands that Arista used, Cisco and its amici have zoomed in too far.[9]

The District Court zoomed in too far as well when it accepted the user interface of the Cisco programs as the relevant work, Appx3, because the user interface is actually a subset of one or more copyrighted Cisco programs. The CLI elements at issue were, in turn, a subset of the approximately 16,000 Cisco CLI terms and 454,000 Cisco IOS commands of the Cisco programs. *Cisco*, 2016 WL 4440239 at *6. Component parts of works of authorship do not qualify as separate works of authorship. *See, e.g.*, *Garcia v. Google Inc.*, 786 F.3d 733 (9th Cir. 2015) (actress' performance in part of a video was not a separately copyrightable work of authorship from the video in which it was embodied). *See also* COMPENDIUM OF U.S. COPYRIGHT OFFICE PRACTICES, *supra*, at § 503.1(A) ("[A]s a general rule, the Office will not issue separate registrations for the constituent elements or individual components of a work of authorship.").

Cisco also sought to zoom out by amalgamating subsets of elements from multiple works (that is, the twenty-six works it registered with the Copyright

[9] Courts have rejected efforts to treat parts of a work as separate works. *See, e.g., NXIVM Corp. v. Ross Inst.*, 364 F.3d 471, 480-81 (2d Cir. 2004) (rejecting NXIVM's argument that individual modules of its 500-page text were separate works); *Sony Computer Entm't Am., Inc. v. Bleem, LLC*, 214 F.3d 1022, 1028 (Sony screen shot used by Bleem was "an insignificant portion of the complex copyrighted work as a whole").

Office) and claiming the amalgamation as a work of authorship. *See, e.g.*, 4 NIMMER & NIMMER, *supra*, at § 13.03[A][3] (cautioning against such aggregation). It appears that Cisco did not select and arrange the CLI elements used by Arista in the manner that Congress envisioned when it decided that compilations could sometimes be copyrighted. The CLI elements at issue were instead nonliteral elements of Cisco's software in which the CLI terms were embodied or enabled. They do not appear to be a unified whole, but rather an amalgam of different types of subcompilations for performing particular types of operations, some of which appear on a screen, but many of which are simply accepted as inputs. As the District Court noted in denying Cisco's motion for partial summary judgment:

> Cisco has not presented evidence of where Cisco CLI comes from or how and when it was compiled. Although Cisco has copyrights covering its IOS, Cisco does not have a single copyright registration covering the compilation it calls the Cisco CLI. Rather, the Cisco CLI is composed of pieces drawn from 26 different copyright registrations covering Cisco's IOS.

Cisco, 2016 WL 4440239 at *3 (citations omitted). The District Court was not convinced that the Cisco CLI was "a compilation that its author(s) put together rather than a creature of its litigation strategy." *Id*.

The copyright statute extends protection to compilations only insofar as their putative authors have exercised a modicum of creativity in the selection and arrangement of data or other materials such that the compilation "as a whole" is an original work of authorship. 17 U.S.C. § 101 (definition of "compilation"). It

appears that the CLI elements at issue had no independent existence as a compilation until Cisco developed a document setting forth the allegedly infringing elements for purposes of this litigation. *Cisco*, 2016 WL 4440239 at *3-4. To be copyright-protectable, a compilation must be an original work of authorship that existed prior to litigation about it. *Id*. at 3. It is unclear that Cisco's CLI compilation copyright claim satisfies this standard.[10]

Insofar as the jury heard evidence that many of the CLI commands were taken from industry standards, as the District Court clearly articulated they did, Appx9-11, and the jury further heard evidence that dozens of other commands were names of common functions, that still others were logical given the tasks the Cisco programs were designed to perform, Appx11, and further that external factors, including customer expectations, constrained choices for still other command terms, Appx11-12, the jury might reasonably have concluded that so many of the CLI commands at issue were scenes a faire that Cisco's claim of infringement was ultimately unpersuasive. There may simply have been too few

[10] It is, moreover, unclear which of the twenty-six IOS programs were arguably infringed, let alone which of the five types of compilations that the District Court identified as possible bases of Cisco's infringement claim the jury was persuaded that Cisco had established as prima facie infringements. Nor was it clear when the Cisco CLI elements at issue were created. Presumably, some of them date back to the earliest of Cisco's IOS programs, while others were incrementally added over the years. If many of the CLI terms date back to the 1980s, but others were added very recently, there may be no one program as to which the CLIs were original.

non-standard original elements at issue to justify a verdict that a compilation of them as a whole had been infringed.

Such a conclusion would be in keeping with rulings in cases such as *Key Publications, Inc. v. Chinatown Today Publishing Enterprises, Inc.*, 945 F.2d 509 (2d Cir. 1991). Key claimed that the defendant's use of 1500 entries from its directory of Chinese businesses in New York City infringed. The Second Circuit affirmed a finding of non-infringement because the defendant had used numerous different categories to organize its competing directory and had included many additional listings. *Id*. at 515-16. *See also Kregos v. Associated Press*, 3 F.3d 656 (2d Cir. 1993) (defendant's rearrangement of categories for baseball game prediction compilation held not to infringe); *American Massage Therapy Ass'n v. Maxwell Petersen Assoc.*, 209 F. Supp. 2d 941 (N.D. Ill. 2002) (not infringement to copy more than 17,000 names and addresses from plaintiff's registry for defendant's database of more than 73,000 listings of massage therapists).

CONCLUSION

The scenes a faire doctrine is grounds for a viable defense to claims of infringement in computer program cases, including those involving reuses of program commands. On appeal, Cisco has taken too crabbed and narrow a view of the scenes a faire doctrine. Under the rubric of scenes a faire doctrine, Ninth Circuit and other appellate decisions in software copyright cases have taken into

account that copyright does not protect industry standards, common terms, logical elements, and elements constrained by external factors, including market expectations. The jury could reasonably have been persuaded that Cisco's incorporation of many of these types of elements in its CLI undercut its claim of copyright infringement. Moreover, Ninth Circuit precedents support the view that software design elements can become standards over time, so even if the jury relied to some degree on evidence that certain terms had become industry standards after Cisco developed the CLI, its verdict could be sustained on that basis as well. *Amicus* respectfully urges this Court to defer to the jury's verdict.

Dated: December 22, 2017

Respectfully submitted,

/s/ Robert Walker

Robert Walker
Catherine Crump
SAMUELSON LAW, TECHNOLOGY & PUBLIC POLICY CLINIC
University of California, Berkeley, School of Law
334 Boalt Hall (North Addition)
Berkeley, California 94720
(510) 664-4875

Counsel of Record for Amicus Curiae Copyright Law Professor Pamela Samuelson

United States Court of Appeals
for the Federal Circuit

Cisco Systems, Inc. v. Arista Networks, Inc., 2017-2145

CERTIFICATE OF SERVICE

I, Robyn Cocho, being duly sworn according to law and being over the age of 18, upon my oath depose and say that:

On **December 22, 2017** counsel has authorized me to electronically file the foregoing **Brief for Copyright Law Professor Pamela Samuelson as *Amicus Curiae* in Support of Defendant-Appellee and Affirmance** with the Clerk of Court using the CM/ECF System, which will serve via e-mail notice of such filing to all counsel registered as CM/ECF users, including the following principal counsel for the other parties:

Kathleen M. Sullivan
Quinn Emanuel Urquhart & Sullivan, LLP
51 Madison Avenue, 22nd Floor
New York, NY 10010
kathleensullivan@quinnemanuel.com
212-849-7000
Principal Counsel for Appellant

Paper copies will also be mailed to the above principal counsel at the time paper copies are sent to the Court. Any counsel for Amicus Curiae appearing at the time of filing will be served via CM/ECF notice only.

Upon acceptance by the Court of the e-filed document, six paper copies will be filed with the Court within the time provided in the Court's rules.

December 22, 2017

/s/ Robyn Cocho
Counsel Press

CERTIFICATE OF COMPLIANCE WITH TYPE-VOLUME LIMITATION, TYPEFACE REQUIREMENTS AND TYPE STYLE REQUIREMENTS

1. This brief complies with the type-volume limitation of Federal Rule of Federal Circuit Rule 32(a). This brief contains 6,726 words, excluding the parts of the brief exempted by Federal Rule of Appellate Procedure 32(f).

2. This brief complies with the typeface requirements of Federal Rule of Appellate Procedure 32(a)(5) and the type style requirements of Federal Rule of Appellate Procedure 32(a)(6). This brief has been prepared in a proportionally spaced typeface using Microsoft Word 2010 in 14pt Century Schoolbook.

Dated: December 22, 2017

/s/ Robert Walker
Robert Walker